CLAIMS

What is claimed is:

1. A system comprising:

a first node operative to provide a source broadcast requesting data, the first node associating an F-state with a copy of the data in response to receiving the copy of the data from memory and receiving non-data responses from other nodes in the system, the non-data responses including an indication that at least a second node includes a shared copy of the data, the F-state enabling the first node to serve as an ordering point in the system capable of responding to requests from the other nodes in the system with a shared copy of the data.

2. The system of claim 1, wherein the non-data responses further comprise an indication that the other nodes in the system do not have a copy of the data requested by the first node.

3. The system of claim 1, wherein the source broadcast requesting the data comprises a non-ownership request for the data.

4. The system of claim 3, wherein the non-ownership request comprises a source broadcast read request.

5. The system of claim 1, wherein the first node comprises a first processor having an associated cache that comprises plurality of cache lines, one of the cache lines having an address associated with the copy of data received from memory and state data that defines the state of the data stored in the one of the cache lines.

6. The system of claim 5, wherein the first processor further comprises a cache controller that controls the state of the data stored in the plurality of cache lines.

7. The system of claim 6, wherein the cache controller is capable of silently evicting the data stored in the one of the cache lines by modifying the state information from the F-state to an invalid state for the data.

8. The system of claim 1, wherein each node defines a processor having an associated cache that comprises a plurality of cache lines, each cache line having a respective address that identifies associated data and state information that indicates a state of the associated data for the respective cache line, each of the processors being capable of communicating with each other via an interconnect.

9. The system of claim 8, further comprising a cache controller associated with each cache for managing data requests and responses for the respective cache.

10. The system of claim 9, wherein the cache controller further comprises a state engine capable of silently evicting data stored in a cache line having the F-state by modifying the state information for the cache line from the F-state to an invalid state for the data.

11. The system of claim 1, wherein the system implements a source broadcast protocol to process requests and responses provided by nodes within the system, the system transferring to an associated forward progress protocol in response to a request failing in the source broadcast protocol.

12. The system of claim 11, wherein the forward progress protocol comprises a directory-based protocol.

13. The system of claim 1, wherein the ordering point defined by the F-state migrates from the first node to another node in response to the another node issuing a source broadcast non-ownership request for a copy of the data.

14. A multiprocessor network comprising:

a plurality of processor nodes in communication with each other;

at least a first node of the plurality of processor nodes includes a copy of data associated with a given address that is also shared with memory, the first node operating in a first state that causes the first node to respond to a non-ownership request from a second node of the plurality of processor nodes for the data by (i) sending a response to the second node that includes a shared copy of the data, and (ii) transitioning from the first state to a second state indicating that the data is shared; and

the second node transitioning to a third state in response to receiving the shared copy of the data from the first node, such that the second node becomes an ordering point in the network for providing a shared copy of the data.

15. The network of claim 14, wherein each of the plurality of processor nodes further comprises an associated cache that comprises a plurality of cache lines, each cache line having a respective address that identifies associated data and state information that indicates the state of the associated data for the respective cache line.

16. The network of claim 15, wherein a cache line in one of the first and second states being capable of silently evicting associated data by modifying the state information for the cache line to an invalid state.

17. The network of claim 14, wherein the network implements a source broadcast protocol to process requests provided by nodes within the network and, if a request fails, the requests are reissued by the nodes using an associated forward progress protocol.

18. The network of claim 17, wherein the forward progress protocol comprises a directory-based protocol.

19. The network of claim 14, wherein the third state and the second state are the same.

20. A computer system, comprising:

a plurality of processors comprising:

a source processor that issues a broadcast request for desired data while in a first state; and

at least one target processor having an associated cache that includes a shared copy of the desired data, the at least one target processor responding to the broadcast request with a response indicating that the at least one target processor includes the shared copy of the desired data;

memory storing the desired data, the memory responding to the broadcast request with a response that includes a copy of the desired data; and

the source processor transitioning from the first state to a second state in response to receiving the responses from the memory and the at least one target processor, the second state enabling the first processor to respond to requests from other of the plurality of processors with a copy of the desired data.

21. The system of claim 20, further comprising at least one other processor having an associated cache that does not include a valid copy of the desired data, the at least one other processor responding to the broadcast request with a response indicating that the at least one other processor does not include a valid copy of the desired data.

22. The system of claim 20, wherein the source processor, after transitioning to the second state, is capable of silently evicting the desired data by returning to the first state.

23. The system of claim 20, wherein the broadcast request for the desired data comprises a non-ownership request.

24. The system of claim 23, wherein the non-ownership request comprises a source broadcast read request.

25. The system of claim 20, wherein the system implements a source broadcast protocol that defines rules for processing broadcast requests provided by processors within the system and, if a request fails, the system transfers to an associated forward progress directory-based protocol.

26. A system, comprising:

means for broadcasting from a first node a non-ownership request for data;

means for indicating that at least one other node in the system has a shared copy of the requested data;

means for providing from memory a copy of the requested data to the means for broadcasting; and

means for enabling the first node to respond to subsequent non-ownership requests for the data from other nodes in the system by providing a shared copy of the data received from memory.

27. The system of claim 26, wherein the means for enabling defines an ordering point in the system for responding to non-ownership requests for the data, the system further comprising means for migrating the ordering point from the first node to another node in the system in response to a non-ownership request for the data provided by the another node.

28. The system of claim 26, wherein the system employs a source broadcast protocol that defines rules for processing broadcast requests provided by processors within the system, the system further comprising means for transferring to an associated forward progress directory-based protocol for processing a request if the request fails in the source broadcast protocol.

29. The system of claim 26, wherein the memory comprises a home node for the requested data, the system further comprising means for blocking the home node from responding with data to another request if the first node provides a response to the another request that includes a shared copy of the data.

30. A method comprising:

broadcasting a read request for data from a source node to other nodes of an associated system;

transitioning the source node into an F-state in response to receiving the data from memory and receiving non-data responses from other target nodes in the system indicating that the data is shared with at least one of the other target nodes; and

enabling the source node, while in the F-state, to serve as an ordering point that is capable of responding to non-ownership requests for the data by providing a shared copy of the data.

31. The method of claim 32, further comprising silently evicting the data from the source node by modifying the state of the data in the source node to an invalid state.

32. The method of claim 30, further comprising moving the ordering point for the data from the source node to another node in response to a non-ownership request for the data provided by the another node.

33. The method of claim 30, wherein the associated system defines a multiprocessor system that includes a plurality of processor nodes, including the source node and the other target nodes, each of the processor nodes comprising a cache that stores data in corresponding cache lines, each cache line having an associated address and state information that defines a state for the data in the corresponding cache line.

34. The method of claim 30, further comprising:

employing a broadcast protocol that defines rules for processing the broadcast read request provided by source node; and

reissuing the read request employing an associated forward progress if the read request broadcast by the source node fails while employing the source broadcast protocol.

35. The method of claim 34, wherein the memory comprises a home node for the data requested by the source node, the method further comprising:

sending an instruction from the source node having the F-state to block the home node from responding with data to a subsequent non-ownership request for the data if the first node provides a response to the subsequent non-ownership request includes a shared copy of the data.